

Polecenie: Szyfrowanie plików. Napisać prosty program, który szyfruje zawartość pliku za pomocą hasła wpisywanego z klawiatury.

Na początku programu zdefiniowałem kilka bibliotek, odpowiedzialnych za długość tablicy, zamianę nazw plików itp. oraz zdefiniowałem wartość MAX-LENGTH-PASSWORD jako 10.

Następnie w funkcji main zdefiniowałem zmienne oraz „uchwyty” do pobrania plików.

Dalej zrobiłem tak jak w poleceniu do zadania, otworzyłem binarnie 2 pliki:

```
if (NULL == (file1 = fopen(argv[1], "rb")))  
return 1;
```

```
if (NULL == (file2 = fopen(tmpname, "wb")))  
return 1;
```

z czego dane z 1 pliku do dane do zaszyfrowania które zapisuje w argumencie funkcji main() jako argv[1] oraz tworzę plik tymczasowy o tymczasowej nazwie używając funkcji tmpnam() tak aby została utworzona całkowicie randomowa nazwa. Podpowiada to delikatny problem przy kompilacji programu. Dokładniej mówiąc pokazuje się ostrzeżenie o tym że:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$ gcc zad1.c -o zad1  
/usr/bin/ld: /tmp/ccDf4NMj.o: in function `main':  
zad1.c:(.text+0x2a): warning: the use of `tmpnam' is dangerous, better use `mkstemp'
```

Ostrzeżenie wyświetla się dlatego że nazwę tego pliku można przypadkiem odgadnąć przez co użytkownik bardzo łatwo dostanie się do gotowego rozwiązania. Natomiast nam nie robi to za dużego problemu ponieważ całość pliku temp zostaje na końcu programu usunięta.

Następnie pobieramy od użytkownika hasło do zakodowania pliku oraz sprawdzamy długość tego hasła.

Na końcu hasło zostaje szyfrowane poprzez funkcję podaną w poleceniu:

```
j = 0;  
while ((znak = getc(file1)) != EOF)  
{  
    znak ^= haslo[j % (dlugosc hasla)];  
    putc(znak, file2);  
    j++;  
}
```

oraz zamieniamy nazwy plików (plik podany przez użytkownika na nazwę pliku temp)

```
rename(tmpname, argv[1]);
```

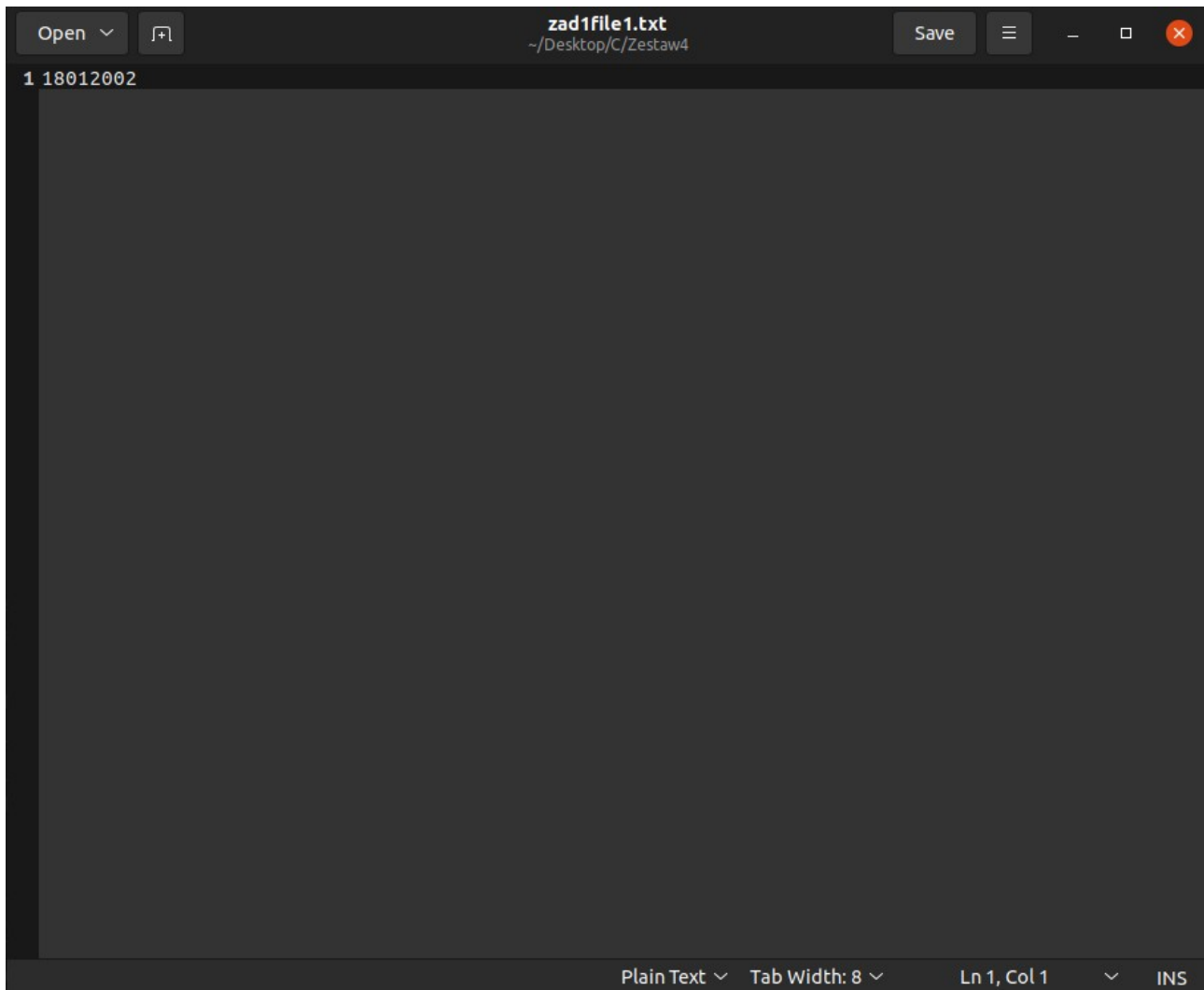
PRZYKŁAD DZIAŁANIA PROGRAMU:

Zaszyfruje 2 pliki o nazwach zad1file1.txt oraz zad1file2.txt (pliki przesyłam dodatkowo w rozwiązaniu zadania). Najpierw skupię się na pliku 1:

=====

PLIK 1 (zad1file1.txt)

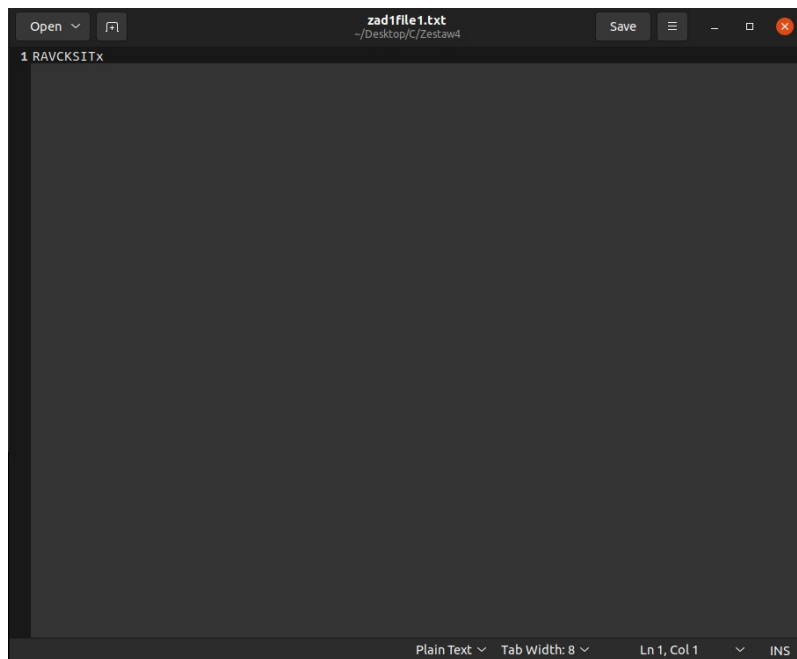
Dane początkowe:



Szyfrowanie pliku:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$ gcc zad1.c -o zad1 -lm -std=c89
/usr/bin/ld: /tmp/ccEjLIWp.o: in function `main':
zad1.c:(.text+0x2a): warning: the use of `tmpnam' is dangerous, better use `mkstemp'
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$ ./zad1 zad1file1.txt
Podaj haslo (max 10 znakow): cyfry
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$
```

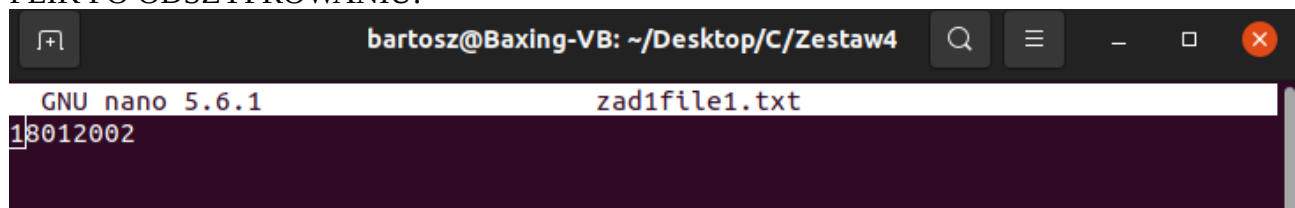
Plik po zaszyfrowaniu:



ODSZYFROWANIE PLIKU:

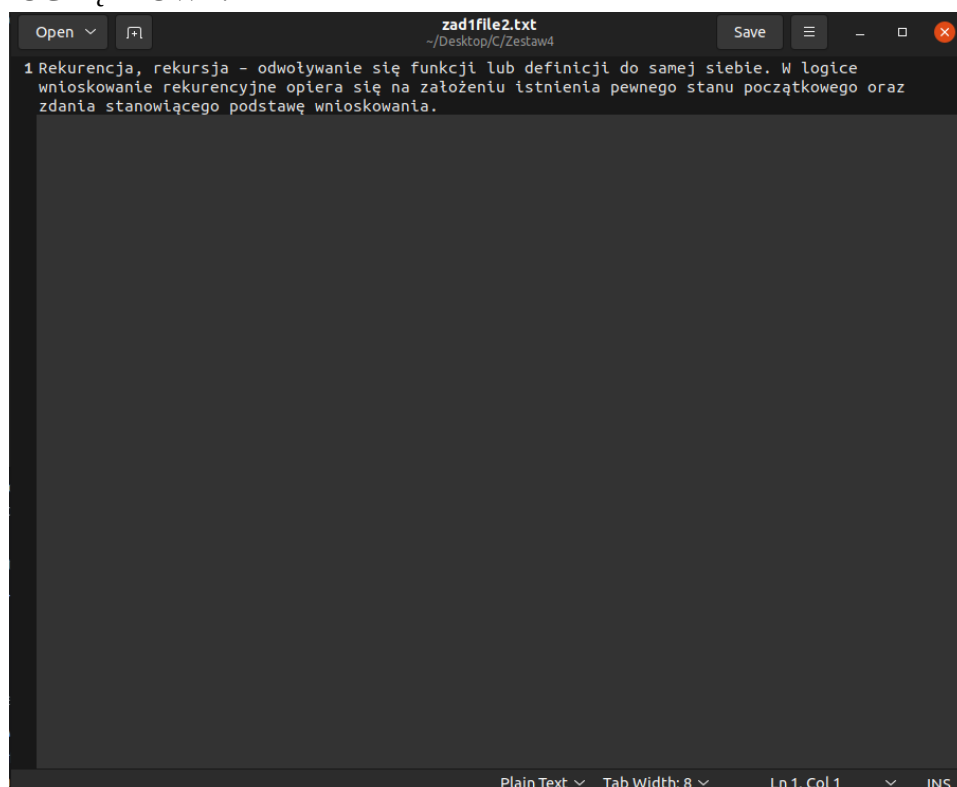
```
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$ ./zad1 zad1file1.txt
Podaj haslo (max 10 znakow): cyfry
bartosz@Baxing-VB:~/Desktop/C/Zestaw4$
```

PLIK PO ODSZYFROWANIU:

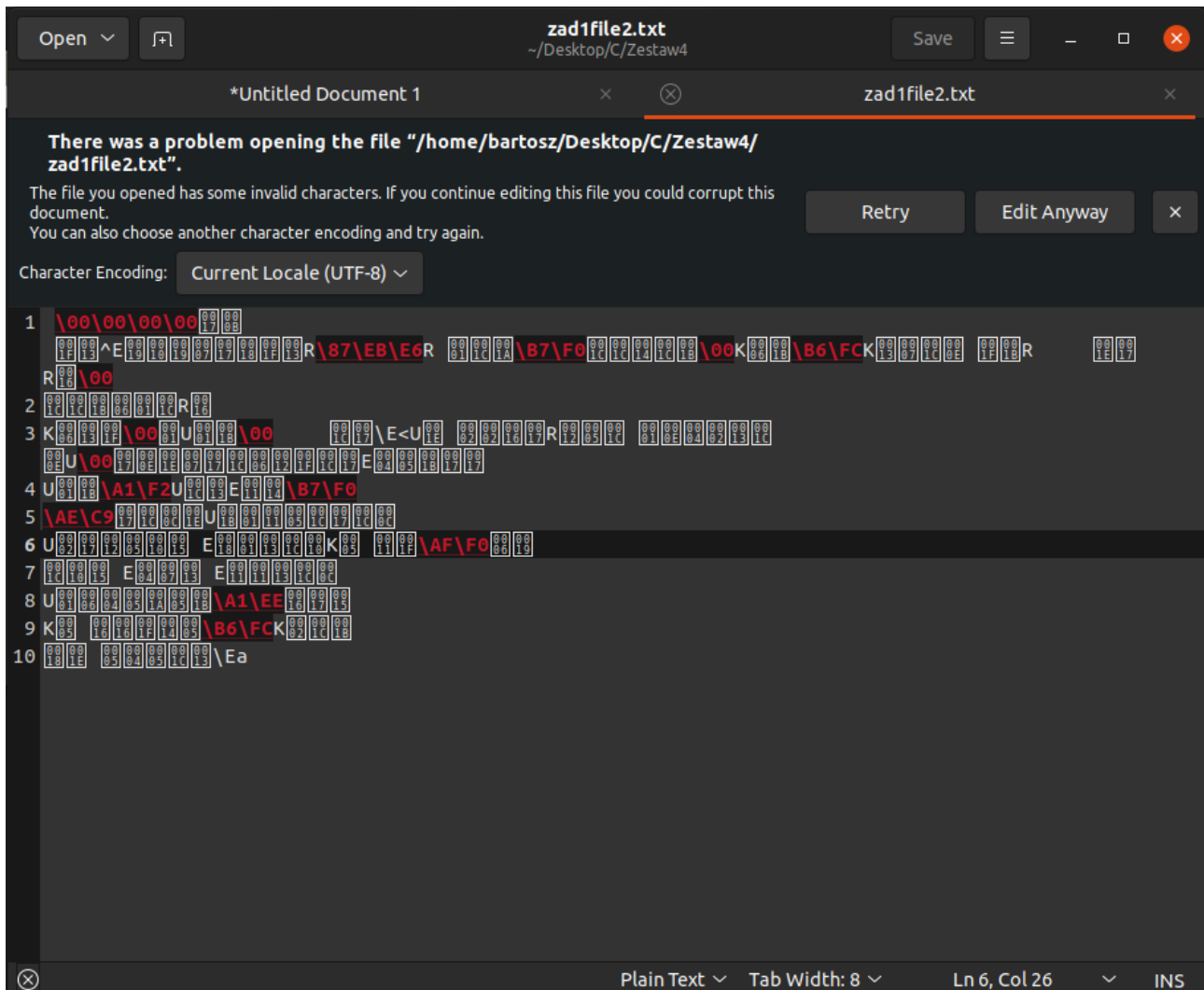


PLIK 2 (zad1file2.txt)

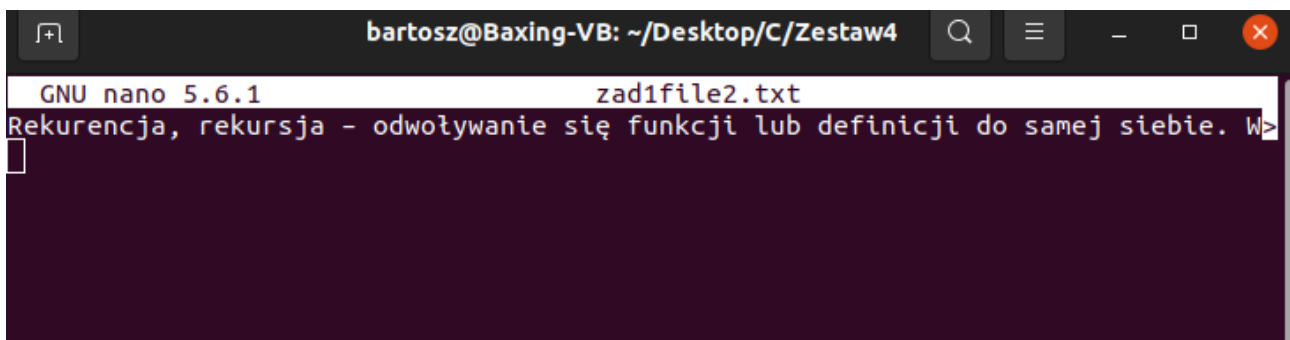
DANE POZATKOWE:



PLIK PO ZASZYFROWANIU:



PLIK PO ODSZYFROWANIU:



Jak widać program działa i szyfruje oraz odszyfrowuje pliki.